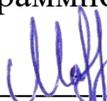


Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Малышев Александр Васильевич  
Должность: Заведующий кафедрой  
Дата подписания: 16.06.2023 13:05:35  
Уникальный программный ключ:  
с44с65fc5eb466e5e378c4db413465be7586с86f

## МИНОБРНАУКИ РОССИИ

Юго-Западный государственный университет

УТВЕРЖДАЮ:  
Заведующий кафедрой  
программной инженерии

  
А.В. Малышев  
(подпись, инициалы, фамилия)

«17» июня 2022 г.

ОЦЕНОЧНЫЕ СРЕДСТВА  
для текущего контроля успеваемости  
и промежуточной аттестации обучающихся  
по дисциплине

Технология разработки программного обеспечения  
(наименование дисциплины)

ОПОП ВО 02.03.03 Математическое обеспечение и администрирование  
информационных систем  
код и наименование ОПОП ВО

# **1 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ**

## ***1.1 ВОПРОСЫ ДЛЯ УСТНОГО ОПРОСА***

*Тема 1. Основные понятия конструирования программного обеспечения.*

1. Что такое конструирование ПО?
2. В чем заключается минимизация сложности программы?
3. Какие существуют методы ожидания изменений в программе?

*Тема 2. Стадия проектирования программного проекта.*

1. Что такое проектирование программы?
2. Какие существуют виды проектирования программ?
3. Что такое архитектура программ?

*Тема 3. Стандарты конструирования программ*

1. В чем заключается стиль конструирования программ?
2. Какие существуют общие стандарты при написании кода?
3. Какие существуют стандарты для языков программирования Python, Java, C#?

*Тема 4. Конструирование программ.*

1. Что такое структура данных? Приведите примеры.
2. Какие существуют основные алгоритмы поиска?
3. Каким образом происходит исключительная ситуация в программном коде?

*Тема 5. Программная документация.*

1. Что такое программная документация?
2. Какие существуют стандарты программной документации?
3. Назовите основные системы для генерации программной документации.

*Тема 6. Системы управления версиями.*

1. Что такое управление версиями программного проекта?
2. Какие существуют средства управления версиями?

3. Назовите причины использования контроля версий.

*Тема 7. Тестирование и отладка программного продукта.*

1. Что такое тестирование программного продукта?
2. Какие существуют виды тестирования?
3. Назовите основные стратегии тестирования программного кода.

*Тема 8. Повторное использование кода.*

1. Что такое стандартная библиотека языка программирования?
2. Какие существуют основные модули в библиотеке NET?
3. Назовите шаги сборки программного проекта.

*Тема 9. Оптимизация кода программного продукта.*

1. Что такое оптимизация?
2. Какие существуют виды оптимизации?
3. Назовите какие бывают настройки компилятора для оптимизации.

**Шкала оценивания:** 5-ти балльная.

**Критерии оценивания:**

Каждый ответ оценивается по дихотомической шкале:

правильно – 1 балл, неправильно – 0 баллов.

Применяется следующая шкала перевода баллов в оценку по 5-балльной шкале:

- 22-27 баллов соответствуют оценке «отлично»;
- □ 17-21 баллов – оценке «хорошо»;
- 12-16 баллов – оценке «удовлетворительно»;
- 11 баллов и менее – оценке «неудовлетворительно».

## **1.2 ВОПРОСЫ ДЛЯ СОБЕСЕДОВАНИЯ**

*Тема 1. Основные понятия конструирования программного обеспечения.*

1. После какого этапа разработки применяется конструирование ПО?
2. Как может измеряться сложность программы?
3. Какие существуют приемы программирования для ожидания изменений в программе?

*Тема 2. Стадия проектирования программного проекта.*

1. Когда применяется проектирование программы?
2. Какие существуют этапы проектирования программ?
3. Какие существуют виды архитектур программ?

*Тема 3. Стандарты конструирования программ*

1. Зачем применяются отступы в программах?
2. Какие существуют главные принципы при написании кода?
3. Чем отличаются стили кодирования в языках программирования Python, Java, C#?

*Тема 4. Конструирование программ.*

1. Какие существуют основные структуры данных? Приведите примеры.
2. Какие существуют основные алгоритмы сортировки?
3. Каким образом происходит обработка ошибок в программном коде?

*Тема 5. Программная документация.*

1. Зачем применяется программная документация?
2. Как программная документация выражается в исходном коде?
3. Каким образом можно извлечь программную документацию из исходного кода?

*Тема 6. Системы управления версиями.*

1. Зачем применяется управление версиями программного проекта?
2. Какие существуют основные функции в системах управления версиями?
3. Что такое удаленное хранилище версий?

*Тема 7. Тестирование и отладка программного продукта.*

1. Зачем проводится тестирование программного продукта?
2. Какие существуют основные способы тестирования?
3. Назовите основные отличия тестирования программного кода в методах черного и белого ящиков.

**Шкала оценивания:** 5-ти балльная.

**Критерии оценивания:**

Каждый ответ оценивается по дихотомической шкале:

правильно – 1 балл, неправильно – 0 баллов.

Применяется следующая шкала перевода баллов в оценку по 5-балльной шкале:

- 18-21 баллов соответствуют оценке «отлично»;
- □ 14-17 баллов – оценке «хорошо»;
- 10-13 баллов – оценке «удовлетворительно»;
- 9 баллов и менее – оценке «неудовлетворительно».

### ***1.3 ПРОИЗВОДСТВЕННЫЕ ЗАДАЧИ***

*Тема 2. Стадия проектирования программного проекта.*

#### *Производственная задача №1*

Напишите программу которая выводит в консоль содержимое текстового файла. Содержимое файла должно отображаться в точности, как в файле, включая табуляции и переводы строк. Имя файла задается в первом аргументе командной строки.

#### *Производственная задача №2*

Напишите программу – калькулятор вводимых вычислений. Программа считывает из консоли арифметическое выражение в виде строки и выводит результат в консоль. Выражение может включать в себя целые и действительные числа, операции сложение, вычитание, умножение, деление и скобки. В программе нужно спроектировать модули лексического и синтаксического анализатора. Вычисления следует производить сразу на этапе синтаксического анализа.

#### *Производственная задача №3*

Напишите программу, которая создает байт-код для интерпретации вычисления арифметических выражений. Выражение может включать в себя целые и действительные числа, операции сложение, вычитание, умножение, деление и скобки. Необходимо спроектировать байт-код интерпретатора. Исходную программу вычислений необходимо передавать в виде текстового файла (в

аргументах программы), выходной файл – двоичный, который содержит байт-код. Имя выходного файла задается в аргументах программы.

#### *Тема 4. Конструирование программ.*

##### *Производственная задача №1*

Напишите программу-интерпретатор байт-кода арифметических и алгебраических вычислений. Вычисления следует проводить, используя регистр аккумулятора и стек операндов. Интерпретатор должен содержать также память для переменных. Операции интерпретатора должны включать: загрузку константы, загрузку переменной, запись значения аккумулятора в стек, выгрузка значения из стека, запись в переменную, сложение, вычитание, умножение, деление, сравнение, конъюнкция, дизъюнкция, отрицание. Результат вычислений выводится в виде значений всех использованных переменных.

##### *Производственная задача №2*

Напишите программу-компилятор. Исходный язык должен включать переменные типа байт и слово, арифметические и логические выражения, операторы присваивания, условий и циклов. На выходе программы – файл с байт-кодом для интерпретации.

##### *Производственная задача №3*

Напишите программу-виртуальную машину. Она проводит вычисления, используя регистр аккумулятора и стек операндов. Управляющие команды используют стек вызовов. Интерпретатор должен содержать также память для переменных. Операции интерпретатора должны включать: загрузку константы, загрузку переменной, запись значения аккумулятора в стек, выгрузка значения из стека, запись в переменную, сложение, вычитание, умножение, деление, сравнение, конъюнкция, дизъюнкция, отрицание, условный переход, безусловный переход, вызов подпрограммы, возврат из подпрограммы, ввод, вывод, останов. Виртуальная машина выполняет заданную в байт-кодах программу, взаимодействуя с пользователем через консоль.

## *Тема 7. Тестирование и отладка программного продукта.*

### *Производственная задача №1*

Напишите тесты для лексического анализатора. Должны проверяться лексемы: целое число, вещественное число, идентификатор, арифметическая операция. Также необходимо включить тесты для проверки игнорирования пустот (пробелов, табуляций, переводов строк) и комментариев.

### *Производственная задача №2*

Напишите тесты для синтаксического анализатора. Здесь должны быть проверены оператор присваивания, условный оператор, оператор цикла, выражение.

## *Тема 8. Повторное использование кода.*

### *Производственная задача №1*

Напишите программу с графическим интерфейсом пользователя для калькулятора. Программа должна использовать платформу .NET.

## *Тема 9. Оптимизация кода программного продукта.*

### *Производственная задача №1*

Проведите измерение производительности для программы-интерпретатора. Определите функции программы с наибольшим суммарным временем выполнения. Оптимизируйте функцию с наибольшим временем выполнения.

***Шкала оценивания:*** 5-ти балльная.

### ***Критерии оценивания:***

оценка **«отлично»** выставляется обучающемуся, если задача решена в полной мере, в программе выполняются все необходимые функции, программа работает корректно на всех тестовых входных данных;

оценка **«хорошо»** выставляется обучающемуся, если задача решена, в программе выполняются большинство функций, программа работает правильно на всех тестовых входных данных;

оценка **«удовлетворительно»** выставляется обучающемуся, если задача решена не полностью, в программе выполняются часть заданных функций, программа работает правильно на всех тестовых входных данных;

оценка «неудовлетворительно» выставляется обучающемуся, если в программе реализованы часть функций, программа работает неправильно на некоторых тестовых входных данных.

## **2 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ**

### **2.1 БАНК ВОПРОСОВ И ЗАДАНИЙ В ТЕСТОВОЙ ФОРМЕ**

Вопросы в закрытой форме.

1. Как найти квадратный корень из числа  $x$ :

- `Math.Sqrt(x)`
- `Summ.Koren(x)`
- `Arifmetic.Sqrt`
- `Sqrt(x)`

2. Обозначения оператора «НЕ» значения оператора «НЕ»:

- `!`
- `No`
- `!=`
- `Not`

3. Обозначение оператора «ИЛИ»:

- `Or`
- `||`
- `!`
- `!=`

4. Чему будет равен  $c$ , если `int a = 10; int b = 4; int c = a % b`

- 11
- 3
- 1
- 2

5. Чему будет равен  $c$ , если `int a = 10; int b = 4; bool c = (a == 10 && b ==`

4):

- True
- Null
- False
- 14

6. Чему будет равен c, если `int a = 10; int b = 4; bool c = (a == 10 && b == 4)`:

- True
- Null
- False
- 14

7. Чему будет равен c, если `int a = 0; int c = a—`:

- -1
- Null
- 0
- 1

8. Что вернет функция `Termin` после выполнения. Код:

```
int Termin(){
int a = 1;
int b = 3;
if (a != 5) return a + b;
else return 0;
}:
```

- 5
- 0
- 3
- 4

9. Какой оператор возвращает значение из функции?

- return
- out

- ref
- foreach

10. Какой класс из System.IO можно использовать для чтения текстового файла?

- StreamReader
- StreamWriter
- BinaryReader

11. Что будет выведено на консоль?

```
Console.WriteLine("Text" + " test");
```

- String.Empty
- Возникнет ошибка
- Text
- Text test
- " test"

12. Что будет выведено на консоль?

```
Console.WriteLine(11 / 2);
```

- 5
- 5.5
- 6

13. Что будет выведено на консоль?

```
1 bool isValid = 1;  
2 Console.WriteLine(isValid);
```

- true
- false
- 1
- Возникнет ошибка

14. Что будет выведено на консоль?

```
public static void Main(string[] args) {  
    {  
    int digit = 5;  
    }  
    Console.WriteLine(digit);  
    }  
}
```

- Возникнет ошибка
- 0
- null
- 5
- ""

15. Что будет содержать строка name после выполнения данного кода?

```
string name = "Alexey";  
name[1] = ".";
```

- "Alexey"
- Возникнет ошибка
- ".lexey"
- "A.ехey"

16. Какой тип переменной используется в коде: int a = 5

- "1 байт\*"
- Целое 32 бит
- Целое 16 бит
- Целое 64 бит

17. Что делает оператор «%»

- Тригонометрическая функция
- Остаток от деления
- Процент от суммы
- Другое

18. Как сделать инкрементацию числа?

- ++
- --
- %%
- !=

19. Верно ли, что тип `string` хранит набор символов Unicode?

- Верно
- Неверно

20. Алгебра логики - это...

- раздел математической логики, значение всех элементов (функций и аргументов) которой определены в двухэлементном множестве: 0 и 1.
- раздел математической логики, значение всех функций которой определены в двухэлементном множестве: 0 и 1.
- раздел математической логики, значение всех аргументов которой определены в двухэлементном множестве: 0 и 1.
- раздел математической логики, значение всех элементов которой определены в двухэлементном множестве: 0 и 1.

21. Естественная форма представления двоичных чисел - это...

- форма с фиксированной запятой.
- форма с плавающей запятой.
- форма с плавающей точкой.
- форма с указанием мантиссы основания и порядка.

22. Инверсия - это операция...

- объединения.
- склеивания.
- отрицания.
- свертки.

23. Информация в компьютере не кодируется в

- двоичной системе счисления;
- двоично-десятичной системе счисления;

- десятичной системе счисления.

24. Какое основание имеет двоичная система счисления?

- 4
- 2
- 6
- 8

25. При переводе смешанного числа следует...

- последовательно делить его на основание  $P$  до тех пор, пока очередная целая часть частного не окажется равной 0.
- переводить его целую и дробную части отдельно.
- последовательно умножать его на основание  $P$  до тех пор, пока очередная дробная часть произведения не окажется равной 0 или не будет достигнута нужная разрядность дробной части числа.
- переводить его целую и дробную части одновременно .

26. Символ  $V$  используют для операции логического...

- вычитания.
- умножения.
- деления.
- сложения.

27. Символ  $\wedge$  используют для операции логического...

- вычитания.
- умножения.
- деления.
- сложения.

28. Система счисления - это

- способ представления информации с помощью символов.
- способ наименования и изображения чисел с помощью символов, имеющих определенные количественные значения.
- способ представления количественной информации с помощью символов.

- способ представления количественной информации с помощью символов, имеющих определенные количественные значения.

29. Алгоритм - это

- упорядоченная последовательность команд, подлежащая обработке.
- конечный набор предписаний, подлежащий выполнению.
- последовательность битов в формате, имеющая определенный смысл.
- конечный набор предписаний, определяющий решение задачи посредством конечного количества операций.

30. Программа для ЭВМ - это

- упорядоченная последовательность команд, подлежащая обработке.
- конечный набор предписаний, подлежащий выполнению.
- последовательность битов в формате, имеющая определенный смысл.
- конечный набор предписаний, определяющий решение задачи посредством конечного количества операций.

31. Слово - это

- последовательность битов в формате, имеющая определенный смысл.
- последовательность, состоящая из определенного, принятого для данной ЭВМ числа байтов.
- структурная единица информации, закодированная двоичными цифрами "0" и "1".
- это имеющий имя информационный массив.

32. Файл - это имеющий имя информационный массив

- последовательность битов в формате, имеющая определенный смысл.
- последовательность, состоящая из определенного, принятого для данной ЭВМ числа байтов.
- структурная единица информации, закодированная двоичными цифрами "0" и "1".
- это имеющий имя информационный массив.

33. Формат - это

- последовательность битов, имеющая определенный смысл.
- последовательность, состоящая из определенного, принятого для данной ЭВМ числа байтов.
- структурная единица информации, закодированная двоичными цифрами "0" и "1".
- это имеющий имя информационный массив.

34. В виде каких двух разновидностей выполняются трансляторы?

- шифраторы и дешифраторы;
- полусумматоры и сумматоры;
- интерпретаторы и компиляторы;
- компиляторы и модуляторы.

35. Исходный модуль перед исполнением должен быть переведен на внутренний язык машины. Какой специальной программой выполняется эта операция?

- транслятором;
- дешифратором;
- сумматором;
- шифратором.

36. Какие коды используются для защиты информации от искажений?

- арифметические коды;
- помехозащищенные коды;
- коды формы;
- цифровые коды аналоговых величин.

37. Какие коды определяют, как должна выглядеть обрабатываемая в ЭВМ информация при отображении?

- арифметические коды;
- помехозащищенные коды;
- коды формы;
- цифровые коды аналоговых величин.

38. Какой тип интерфейса может иметь операционная система ПК?

- командный интерфейс;
- интерфейс в виде текстовых меню;
- интерфейс в виде графических меню;

39. Системное ПО включает в себя:

- операционные системы и сервисные программы;
- операционные системы и инструментальные программные средства;
- операционные системы и систему технического обслуживания;

40. Что не относится к прикладному программному обеспечению ПК?

- графические редакторы;
- операционные системы;
- системы управления базами данных;
- информационно-поисковые системы.

41. Что относится к инструментальным средствам?

- трансляторы;
- загрузчики;
- средства отладки;

**Шкала оценивания результатов тестирования:** в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения составляет 36 баллов, по очно-заочной и заочной формам обучения – 60 баллов (установлено положением П 02.016).

Максимальный балл за тестирование представляет собой балл по промежуточной аттестации для данной формы обучения (36 или 60).

Соответствие 100-балльной и пятибалльной шкал:

<i>Сумма баллов по 100-балльной шкале</i>	<i>Оценка по дихотомической шкале</i>
100-85	отлично

84-70	хорошо
69-50	удовлетворительно
49 и менее	неудовлетворительно